Abstract

The paper discusses briefly the evolution of library management systems (LMS) and how changes in technology, information environment, user expectations and searching behaviours, competition from related application streams and the availability of enterprise-wide systems particularly in academic and research environments have influenced changes in LMS functionality and design. The drawbacks of current LMS offerings, both commercial and open source, are then described followed by a description of major new initiatives that have taken place in the last two or three years leading to new ways of freeing the LMS from its monolithic nature into one which supports new workflows via services-oriented architectures (SOA) and web services. These initiatives, particularly that of the OLE Project, eXtensible Catalog, the recommendations of the Digital Library Foundation (DLF) and National Information Standards Organization (NISO), and the recent proposal of OCLC to move LMS into a web-spaced platform using cloud computing paradigms are discussed.

Introduction

Library Management Systems (LMS) or computer-based systems that automate one or all functional areas of a typical library have had a history of evolution going back to the mid 1950s. LMS have also been referred to as Integrated Library Systems (ILS) in later years to reflect the fact that all functions are managed via a central database (what is today being called a siloed application) with processes that transparently exchange data between functional components such as catalogue records and circulation transactions. This paper examines current initiatives that will determine the future of LMS. To understand and appreciate these initiatives it is important to briefly look at the past and recount the influences that have played a role in the evolution and how new influences both within libraries and outside have made it necessary to rethink the design of LMS.

A Snapshot of the Evolution of LMS

The paper discusses the drawbacks of current commercial and open source LMS and the need for new design principles that take advantage of new software and interoperability paradigms such as services-oriented architecture (SOA) and web services that have arisen from the distributed nature of the web, changing user behaviours and the need to manage both core functions of a traditional LMS, new electronic resources plus the capability for interoperating with external applications, e.g., course management systems, personnel directory systems, that are now becoming an integral part of institutions. Initiatives of the OLE Project, the eXtensible Catalog Project, the proposals of the Digital Library Foundation (DLF), the National Information Standards Organization’s (NISO) proposals for best practices and OCLCs recent proposal to use cloud computing paradigms to move the traditional LMS to becoming a fully web-spaced one (as opposed to just web-based) are discussed as pointers to the emerging future of LMS.
Middle generation systems (1960s – 1970s)
- Metadata standard for bibliographic records (MARC) became available;
- Emphasis was on exchanging bibliographic data, centralized cataloguing and distribution of catalogue cards;
- Systems were developed by vendors which leveraged the catalogue data in other modules – circulation, acquisitions;
- First generation integrated LMS came into being;
- These were targeted to single libraries;
- Proprietary backend designs (e.g., flat files) were common; and
- Mostly mini-computer based; character-based interfaces; some systems were still home-grown.

Pre-Internet generation (1970s – up to 1990s)
- Networking via LANs and WANs became possible and libraries began to ask for networking of closely related libraries;
- Microcomputer-based systems with richer interfaces;
- Client-server LAN systems became the norm;
- Interactive applications became possible with GUIs;
- Vendor systems with networking capabilities became available;
- Marketplace soon made home-grown systems unnecessary and not cost effective;
- Most integrated systems had similar functionality with small differences;
- First generation OPACs made their experience. The OPACs were heavily librarian-centric in design; and
- Federated searching became possible via the z39.50 Information Retrieval protocol; and
- Movement away from proprietary to RDBMS-based backend systems and SQL-based search systems.

Internet generation (Web 1.0) (1990s – 2000)
- Initial move was to host the OPAC on a web server; other functional modules were still locally administered;
- Rich GUI front ends using tools like Visual Basic, Visual C++ became available;
- When reliable Internet connectivity became widely and cheaply available in the 1990s, new client server systems that used the web for data storage and transaction processing became available;
- Platforms like JAVA and .NET became the development options for web applications;
- Open source OS platforms like Linux made an entry. Few applications and quite geeky; and
- Backends were still predominantly RDBMS-based and search systems were SQL-based.

Post 2000 – the Web 2.0 Era

The Web became the platform of choice for software. Development philosophies changed from finished product to work-in-progress and frequent updates delivered over the web;
- The web has become from an information delivery only platform to a participative platform. Ordinary individuals contributed via blogs, wikis, podcasts and social networks. This has impacted the expectations that library users have from libraries and LMS;
- Web services via protocols and APIs resulting in information reuse, greater interoperability, RSS/Atom feeds, mashups enhanced user experience in discovery applications, e.g., Amazon, LibraryThing;
- Open source offerings make a serious entry into the marketplace;
- Dissatisfaction with the monolithic nature of the LMS and the OPACs is increasingly voiced;
- The consolidations and mergers in the commercial marketplace is evidence of upheavals in the industry;
- New kinds of enterprise applications have become available to institutions and there is demand for better integration of LMS with such systems.

The snapshot overview of the evolution can also be seen from the point of developments in technology, e.g. changes from using mainframe to mini-computers to microcomputers; from software for un-integrated systems to integrated systems; from single library systems to multi-library and networked systems; from using proprietary to relational database backends; and from LAN-based systems to web-based systems. Developments in both hardware and software technology and the use of new paradigms such as the relational model, object-oriented analysis and design, client-server architectures and languages particularly well-suited to the world wide web have had an influence on the evolution. A major technological influence has been the growth of the web and its distributed environment under different platforms, formats, languages and data models requiring that the LMS supports interoperability.

Equally important influencing factors that have challenged LMS with new demands from librarians as well as users have been:
• **Changes in the information environment**

The emergence of new forms of information, e.g., the web page, electronic forms of conventional information objects such as audio and video, full-text, e-serials. The plethora of formats in which information objects could occur (e.g., in proprietary ones (MS-Windows-based) or as open formats such as HTML, XML, PDF, open document format, MP3, MP4, WMV, JPG, TIFF, etc.) have also required that LMS should be able to deal with new information objects.

• **Changes in user behaviours and demands**

This has probably been most challenging of all influences on the evolution of the discovery interface or OPAC built into LMS. Some of the searching and use behaviours that have challenged LMS are:

- Users want greater freedom in managing their access to information.
- Users want access not only to just library-held information but to other material types and on the web in general.
- Users seek a simple search interface that is not only easy to use but also retrieves items ranked by relevance and points to related items, reviews, recommendations, and allows a degree of faceted searching.
- Users want access to full-text and other digital content and expect the library to assist them in obtaining the full text or other digital content via the LMS.
- The Google generation (the teens of today, who have been brought up on the web and its resources unlike their predecessors) demand the freedom to tag items of their interest, access to information by their own tags or those of peers in a social network. They also value access to reviews, recommendations, and peer ratings of materials that may be useful to them.

• **Development of new metadata standards and protocols**

Although the MARC metadata standard has been a long-standing one for bibliographic records, its complexity and the need for a high level of training for its use to create metadata records is a shortcoming in its use by non-librarians, e.g., authors, painters, musicians, social activists who are today also generators of information. These require to be described in institutional and web-based search systems including generic search engines. This has led to the development of simpler and more generic metadata schemas such as Dublin Core. Other information objects, e.g., courseware and learning objects require metadata that is not covered well enough by bibliographic standards. It is important to recognize that today’s users, particularly in the academic world, require access to other materials as well and they expect that the LMS should be able to inter-operate with such systems in meaningful ways. The open access movement and the development of the OAI-PMH has enabled the development of institutional archives of scholarly contributions. These are valued by researchers and faculty and there is demand for the interoperability of such resources with the LMS.

• **Emergence of related application streams leading to pressures from librarians, end-users as well as Institutions**

Database producers, e-journal publishers, providers of data, audio and video feeds and content, subject portals, learning management systems, enterprise-wide information systems have their own workflows, search interfaces, applications and metadata standards. There is a growing demand from librarians, users and institutional heads that libraries should interoperate their systems with these related applications to permit access to a wider information base and to avoid unnecessary duplication of similar data across applications and avoidable errors in transactions that may take place between LMS and other applications, e.g., between a LMS’s acquisitions system and the Institution’s Purchase Management System.

**Limitations of Current Crop of LMS in Today’s Context**

One of the advantages of current offerings of LMS is that it tightly integrated all functions within a common application as a means of increasing efficiencies. However, what was once considered to be a virtue, has many drawbacks in the changed times of today. Some of the drawbacks are:

- The LMS is a complex, closed system, the software uses proprietary code and is expensive to license and difficult to customize even if the software is open source. The complexity of the code militates against customization by a third party. Even if this is theoretically possible, it is expensive in terms of development costs.
- The LMS imposes rigid workflows. These are suitable for conventional materials. The management of electronic resources requires different workflows, e.g. management of: digital rights, management of access rights to e-journals, implementing consortial borrowing, document delivery and access to full text via applications (e.g. openURL) and protocols. Libraries are faced with two options: either to use the inefficient workflows to manage e-resources with their LMS or to implement a parallel system for the management of electronic resources. Parallel systems are obviously an additional burden in terms of costs and maintenance.
• New enterprise-wide information systems, personnel directory systems and purchase management systems are being implemented. Current LMS do not integrate with the new systems. Libraries create complicated processes for extracting data from the enterprise systems, reprocess data inside the LMS, and then send data back to the enterprise systems, e.g. student or patron data; library acquisitions data

• Lack of integration with widely used tools, e.g. database search systems, institutional repositories is a serious deficiency. Libraries cope with these problems by developing add-on components or by purchasing new LMS components and writing programs to connect them to them to the LMS

• It is nearly impossible for a library to integrate its commercial ILS with tools outside the LMS, such as a course/learning management system or social-networking tools.

• Current OPAC offerings of LMS, most of which are librarian-centric do not provide the discovery experience that many users are accustomed to in collateral systems such as Amazon.com, eBay, Google, LibraryThing, social network applications.

• New OPAC offerings in the commercial space, e.g. Endeca, Primo, Aquabrowser, improve user experience, but purchasing and implementing a second OPAC is an extra expense and an extra support burden on top of costs and support for the LMS. New open source OPAC offerings such as Scriblio, VuFind have also become available, but use of these requires programming effort on the part of the libraries and the need for the vendor of the LMS to expose ways in which third party applications can use the data embedded in their application.

• The work done and experience gained to add-on new workflows in existing LMS to cater to the management of newer resources is not easily transferable to other LMS products or to other libraries trying to solve the same problems.

• The OLE Project (www.oleproject.org) under the leadership of the Duke University, USA
• The DLF Discovery Interface (DLF-DI) Task Force
• The NISO Best Practices for Designing Web Services in the Library Context
• The eXtensible Catalog project of the University of Rochester, USA
• The OCLC proposal for a web-scale, cooperative library management service

Of the above initiatives, the OLE Project has had the widest participation (with libraries, vendors, library associations) and must therefore be considered as representing a broad spectrum of views and concerns.

The rest of this paper describes the above-mentioned initiatives in some detail and their projected outcomes.

The OLE Project
The project (www.oleproject.org) is an initiative of the Duke University in the USA. The project, begun in 2008 with funding from the Andrew Mellon Foundation and has involved a multinational group of libraries in the USA, Canada, Australia and UK with the objective of developing the design for an Open Library Environment (OLE), an alternative to the current model of an Integrated Library System. The goal is to produce a design document to inform open source library system development efforts, to guide future library system implementations, and to influence current Integrated Library System vendor products.

The aims of the Project are to:

• Adopt a transformational design approach to create a flexible, interoperable system that goes beyond replicating the concepts that underlie legacy LMS.

• Develop and use open, flexible technology to produce a community-sourced alternative to current LMS and electronic resource management (ERM) systems.

• Automate core library functions in a way suited to modern workflows and one that interoperates with business and content applications beyond the library.

OLE is a framework and platform that goes beyond the monolithic LMS through its ability to utilize other systems and deliver valuable new services. At the same time, OLE allows institutions to avoid redundancy of data, and reduce purchase and integration of add-on components to their current ILS in order to carry out library business. The OLE places the library’s business in context within the fabric of the institution and the research process, rather than keeping it a separate, siloed operation (http://oleproject.org/annual-ole-project-report/). The Australian National Library, one of the partners in the OLE Project,
articulated a service framework based on a services-oriented architecture (SOA).

In the SOA sense of the word, a service is a set of capabilities offered by a service provider through a service definition that makes the capabilities discoverable by a service consumer. The implementation of the service is “opaque” to the consumer, and the invocation of a service itself may seek out the SOA-defined capabilities of another service provider. Importantly, the services may be locally owned or those under the control of others. The articulation of SOA requires service definitions and agreed protocols and schemas (or data models). Thus standards, in this case, web services-based, will be required to be defined. This is where bodies such as the DLF and NISO have made their presence felt.

The OLE framework has also suggested a predominantly SOA-based model and has the following characteristics:

**Flexibility:** Supports a wide range of resources; accessed by a wide range of customers in a variety of contexts; provides structures for extending and adding new types of resources, customers and contexts.

**Community Ownership:** Designed, built, owned, and governed by and for the library community on an open source licensing basis; sustained by the community with the assistance of a thriving vendor marketplace; evolves over time through transparent processes that enable and respond to input and innovation from the community.

**Service Orientation:** Developed using the methods of Service Oriented Architecture (SOA) and implemented with web services to be a modular and technology-neutral framework that ensures the interoperability of library business systems and accommodates a diversity of solutions without the risks posed by single-source providers. Further, it can be customized to support local needs.

**Enterprise-Level Integration:** Designed to adapt to and integrate with other enterprise systems such as research support, student information, human resources, identity management, fiscal control, and repository and content management.

**Efficiency:** Provides a modular application infrastructure that integrates with new and existing academic and research technologies and business processes for improved efficiency and effectiveness of the institution; meets current and future business needs of the community.

**Sustainability:** Creates a reliable and robust framework to identify, document, innovate, develop, maintain, and review the software necessary to further the operation and mission of libraries.

The OLE Project draft final report (http://oleproject.org/ final-ole-project-report) provides an open source reference implementation of a set of technology services that allow libraries to carry out their back-end business operations, and therefore to replace their current ILS. However, OLE will also support functional capabilities that go beyond the existing ILS core. Each OLE component will be highly modular and use standards-based interfaces, allowing an institution to mix-and-match OLE components with other, existing campus and library systems if desired—including both open source and commercial systems. This will allow institutions to install only selected portions of OLE, or to adopt a phased replacement strategy. OLE will not build the front-end or resource-discovery functionality of today’s ILS’s, but will instead provide support for several open source front-end projects (e.g. eXtensible Catalog, VuFind, Blacklight), as well as any commercial alternative that is willing to use OLE’s standards based interfaces. OLE also expects its services to be delivered through non-library interfaces such as smart phones, learning management systems, campus portals, and other institutional products that involve resources and/or support from the library.

The Abstract Reference Model for OLE (below) shows the relationship between OLE middleware, OLE components, entities acted on by OLE, and third-party components, such as Identity Management, Institutional Repositories, and Course Management Systems.

The OLE Reference Model is an abstract representation of the OLE framework. As such, it describes the high-level functional components that will form OLE. Each of these components is made up of a number of workflows and/or processes. A workflow is a series of activities that involve people, business processes, and software that achieve a library business goal. For example, the OLE component Describe Entity is comprised of the processes Obtain Metadata, Create Metadata, Modify Metadata, Delete Metadata and Expose Metadata. Additionally, the reference model shows examples of third-party components with which OLE will interoperate. These are reusable services, not developed or supplied by OLE, that fulfill an OLE business process. The components that straddle the boundary between OLE and third-party components represent the functions that will be provided partly by OLE and partly by third-party components. The reference model also includes the entities that have so far been identified as belonging in the OLE framework. These are resources, collections, persons, organizations, and services. Finally, the bottom portion of the Reference Model illustrates the software that will manage and connect OLE components. It is this middleware that will provide interoperability with third-party applications.

An important point to note here is that the OLE Project used the ideas of the Australian NLA and also aligned...
its framework with a higher level framework such as the framework for Education and Research or e-Framework (www.e-framework.org). Similarly the OLE project aligns itself also with one or more services defined by the DLF. In other words, OLE will use the results of parallel activities both within the public domain as well as those of commercial vendors which conform to the principles of web services.

SOA is an architecture philosophy or framework that breaks down business processes (in an application area such as LMS) into several generic services. Business processes or workflows, a service usage model (SUM) and protocols and schemas are defined. The framework also assumes that there is a Workflow Engine and a Rules/Policy Engine. The Workflow Engine manages the modeled business processes whereas the Rules Engine modifies the workflows based on local policies. In other words a library using the OLE framework can customize the Rules Engine to suit its policy, which then modifies the Workflow Engine suitably for its purpose. This way, there will be no need for a library to reinvent commonly used workflows via a service layer (e.g. in a check-out transaction) but at the same time have the possibility of modifying the workflow with an appropriate change in the Rules Engine. The fact that workflows are defined within generic services, allows that two or more tasks may use the same service layer to accomplish different tasks. This will ensure reusability of software and re-purposing of software, two key advantages of SOA. For instance, the Deliver Entity component (shown above) describes processes that track the request and supply of a resource. It includes processes that initiate and receive the request, identify the user requesting the resource, check and verify the user’s credentials, and determine availability and terms of use of the resource requested. A message is sent to the user whenever a condition is not met. The resource is supplied if all conditions are met.

Another important feature of the OLE Project is that a Build Project is proposed following the planning phase which was completed in July 2009. The Build Project will be a Community Sourced Project. It means a project that will be funded by a consortium of institutions that will sign an agreement and contribute financial/human resources in the building and testing of the The OLE planners have recommended becoming a project within the Kuali Foundation (www.kuali.org). The Kuali Foundation, which directly supports administrative community source software projects in the higher education sector, was designed as a lightweight administrative organization and can provide administrative functions for OLE at a fraction of the cost of building an entirely new entity. In addition, the SOA middleware that Kuali has developed will play an important role in the OLE Build Project. The Build phase of the project is expected to begin soon after the funding proposal is given to and approved by the Andrew Mellon Foundation.

The advantages of building the software as a community sourced project are that it will:
It is not as if only OLE is alone in choosing to go the better fit their needs. and engines. Libraries that don't need certain services and workflows they need through workflow managers and engines. Libraries that don't need certain services can turn them off or adapt them to the workflows that make OLE different from current monolithic LMS products is that the latter emphasize internal functional integration while the former advocates the freeing of the LMS via use of external services and tools (web services, APIs) using commonly agreed business processes, protocols and data models so that a library can choose to use those services, processes, etc., that it requires rather than committing itself to a complex internally integrated product.

Since the release of the final draft report, some vendors have criticized (http://commentary.exlibrisgroup.com/2009/08/ole-unanswered-questions.html) the approach spelt out by the OLE project for contradicting itself: first by stating that the OLE Project was to “… guide further library and to influence current Integrated Library System vendor products” and second by stating that “… a proposal to carry the project forward into the next phase of building the software”. The criticism is that the project has not spelt out the business model that will support development based on the demand, market share expected, the existing competition, risks that institutions would be willing to take in investing in the build project.

In spite of the criticism, the support OLE has received, the meticulousness with which the OLE Project’s planning phase has been conceived and implemented and the fact that over 300 libraries in 4 countries participated in the exercise is indicative of the importance that was attached to the project and the consensus on using a totally different approach to designing and implementing LMS functions and beyond. The impact of OLE is bound to change the face of LMS, whether via the OLE Build Project or via vendor’s adopting the principles and ideas that are undoubtedly sound.

The DLF Discovery Interface Task Force Recommendations

The DLF in 2007 created a ILS Discovery Interface Task Group (ILS-DI) to:

- analyze issues involved in integrating integrated Library systems (ILS’s) and discovery applications, and
- create a technical proposal for accomplishing such integration.

A set of revised recommendations was released in December 2008. The task force recommendations were preceded by a careful analysis of the premises that should drive the task force’s work. The premises include the following:

- OPAC interfaces currently provided by most LMS’s cannot by themselves meet the demands of users in a world where the availability and
sophistication of digital resources and web applications has increased significantly.

- This does not reflect on badly designed interfaces; it reflects the fact that users now need a wider variety of capabilities than any one software package can be expected to provide.

- These trends imply that the ILS needs to become a platform that supports appropriate interfaces for discovery applications living on top of it instead of trying to do everything on its own.

- Some ILS vendors provide proprietary methods for accessing their underlying data stores, e.g. API tools accessible only for a special fee to trained users, or direct SQL queries against the ILS’s database

- These methods do provide the hooks into the ILS but making library resources more widely usable requires a larger, standards-based API.

- To enable outward integration ILS’s should adopt a more standardized method for providing API (or web services) access to the data store

- There is need to move away from traditional library-centric protocols like Z39.50 toward an XML-based web services API model.

- Such a model will enable developers outside of the library community to more easily access the information stored within the ILS, creating opportunities for greater integration with non-library applications, e.g., course management tools.

The task force recommendations comprise a set of Abstract functions and bindings that should be satisfied by APIs and/or protocols at different levels of interoperability.

The work of the DLF task force is for the benefit of developers of open source and commercial LMS vendors and will no doubt be used by the OLE Project in its build project. The future of LMS discovery interfaces will be bound by the standards and protocols that have been articulated by the DLF and a national standards body such as the NISO.

**The NISO Best Practices for Designing Web Services in the Library Context**

In a parallel move but not confined to discovery interfaces, the NISO too has come up with broad principles which may guide the design of web services ([http://www.niso.org/publications/rp/rp-2006-01.pdf](http://www.niso.org/publications/rp/rp-2006-01.pdf)) of the following categories:

- **Discovery services**
  
  Web services to discover metadata, full text or a service; web service to create and maintain a directory (such as a directory of services, a directory of policies, or a directory of members)

- **Locate Services**
  
  Web services to communicate requests and circulation transactions between peer circulation systems, e.g., between members of a consortium of libraries.

- **Requesting Services; Delivery Services; Common Services**
  
  Like the DLF-DI task force recommendations, NISO guidelines are targeted to developers of LMS products. The OLE build project, no doubt, will take notice of the NISO guidelines and to that extent there is a common thread that will bind these apparently independent initiatives in the years to come. Both the DLF and NISO speak of ‘Services’, clearly indicating that the future of library applications will be firmly founded on the principles of SOA.

**The eXtensible Catalog (XC) Project of the University of Rochester**

The XC Project is to design/develop a set of open-source applications that will:

- **Provide libraries with an alternative way to reveal their collections to library users.**

- **Provide easy access to all resources (both digital and physical collections) across a variety of databases, metadata schemas and standards**

- **Enable library content to be revealed through other services that libraries may already be using, such as content management systems and learning management systems.**

- **Make library collections more web-accessible by revealing them through web search engines.**

As can be seen the objectives of the XC as indeed the OLE initiatives spoken of earlier are similar. However, the XC’s uniqueness lies in the fact that it has developed and released various Tool Kits as open source implementations.

The following tool kits have been released:

- **OAI.** This will allow integration of XC with an existing ILS and digital repositories.

- **NCIP.** This tool kit will allow RFID functionality to be enabled within An existing LMS

- **Learning Management.**

The following tool kits are under development:

- **Metadata services – four types of metadata will be available to XC: bibliographic, holding, item and authority data. The tool kit will also have live access to circulation status, authentication mechanisms, and native ILS circulation request functionality.**

- **Drupal (Content Management System) Tool Kit - XC will offer a number of user interfaces including one that is embedded into the Drupal CMS.**
A tool kit that will be embedded into the Blackboard Learning Management System

Toolkits will incorporate unanimously agreed-upon standards for searching through library resources including faceted browsing interface and compliance with the Functional Requirements for Bibliographic Records (FRBR) which lays stress on the usefulness of bibliographic displays rather than just the matching of search terms with catalogue records.

The fact that XC is open source will provide possibilities for customization, although this may be a theoretical advantage since the cost of such customization is not known. The XC Project and its tool kits are already recognized by the OLE Project as resources that can be leveraged in its Build Project plan. To this extent the work will contribute to the future of LMS and open source initiatives in this area.

Web-Scale service proposal of OCLC

In an announcement made by OCLC in 2009, they have proposed to offer “the first Web-scale, cooperative library management service.” This will ultimately, it is hoped, bring into WorldCat Local the full complement of functions traditionally performed by a locally installed integrated library system (ILS) via the web. In this, the OCLC will be using the WorldCat database, undoubtedly the world’s largest bibliographic database of over a billion records plus OCLC’s “cloud,” or bank of servers and communications infrastructure. OCLC’s vision involves shifting increasing portions of activity managed library-by-library through locally or consortially implemented automation to the network level, under its global WorldCat infrastructure. OCLC plans to work with the more than 1,000 libraries and partners that are currently using OCLC library management systems in Europe and Asia Pacific to help build the new service. Like in the full version of WorldCat, WorldCat Local users search against the massive WorldCat.org database, with their local library’s holdings presented first in result lists.

The proposal envisages the use OCLCs’ enriched WorldCat Local with article-level content including the material from its ArticleFirst service plus a vast body of articles from EBSCO to those who subscribe to FirstSearch and EBSCO host. Other providers may be added in course of time. WorldCat’s metasearch enhancement will allow search also of a library’s licensed content. The WorldCat Navigator (an extension of WorldCat Local) will enable consortial borrowing.

OCLC plans to move circulation and acquisitions functions in time.

- circulation is planned to be implemented via a web-based client
- Acquisitions through WorldCat Local will include functionality of the ILS print acquisitions module and an electronic resource management (ERM) system.

These moves could bring the local ILS in use at the libraries which may use this web-scale service, redundant. Since the users of OCLC are among the biggest libraries in the world, their influence on the rest is bound to be significant.

However, there are also concerns about the OCLC proposal:

- a library using the OCLC cloud services route, will be exposing themselves to the risk of giving away to OCLC control of how all their data is used and shared
- The planned OCLC system would probably appeal mainly to smaller- to medium-sized institutions. Large libraries may not like to compromise on functionality and customization to be “web scale,” as OCLC is describes it.
- Developing country libraries, many of whom are not members of OCLC will not find the option appealing
- The plan is presently confined as being applicable to a few ILS vendor systems

Conclusions

The LMS industry is going through a profound transition thanks to the initiatives presented here. The end of an era and the beginning of a new one in the evolution of LMS is seen - from that of a library-specific one to that of an enterprise-wide one.

Vendors are evaluating how to respond. They will not become redundant if they adapt. Commercial and open source offerings incorporating the ideas and work of the initiatives mentioned above among others will, no doubt, become available. Open source initiatives, particularly the Community-Sourced ones, are likely to significantly expand the options for libraries, worldwide. Services to libraries will probably be the next big opportunity rather than products.

References
1. OLE Project overview. http://oleproject.org/overview/
2. The eXtensible Catalog Project. Purpose and goals. http://www.extensiblecatalog.org/about/purpose


